Arhitectura client-server

Arhitectura client-server este o infrastructura software dedicata comunicarii intre doua entitati numite Client si Server, care urmareste stabilitatea, flexibilitata si scalabilitatea.

Modul de comunicare al acestei arhitecturi este bazat pe “messaging pattern” care defineste un set de reguli, limbaje si dialog. Protocolul de comunicare utilizat este TCP care mentine o conexiune deschisa pana cand schimbul de mesaje initiat a fost complet.

Clientul, denumit si front-end, este interfata directa prin care utilizatorul poate realiza o serie de operatii asupra datelor aflate in baza de date, respectand anumite norme de securizare. In functie de operatiile efectuate asupra datelor se pot defini trei tipuri de clienti:

* Client “thick” – se defineste ca fiind un client care executa cele mai multe procesari de date, bazandu-se foarte putin pe server
* Client “thin” - clientul care se bazeaza cel mai mult ca procesarile de date sa fie executate pe server
* Client “hibrid” – combinatia celor doua

Serverul, denumit si back-end, ruleaza unul sau mai multe programe utilizate de toti clientii si care au resurse comune distribuite printre acestia. Tipuri de server:

* Application Server – serverul gazduieste o aplicatie web care poate fi accesata prin intermediul internetului
* Server computational – gazduieste o cantitate mare de resurse distribuite pe internet, necesita o putere mare de procesare si memorie RAM mai mare decat la un calculator normal
* Server baza de date – stocheaza informatiile sub forma unor baze de date si le distribuie in functie de autorizatii diferitelor software-uri
* Web Server – gazduieste pagini web

Serverul poate fi accesat concomitent de mai multi clienti, permitand efectuarea mai multor operatii simultan si gestionarea datelor mult mai eficient decat un client de tip “thick”

Tipurile de arhitectura client-server:

* Two-tier

Clientul si serverul interactioneaza direct, impartind logica aplicatiei, stocarea si procesarea datelor intre cele doua entitati.

De cele mai multe ori este utilizata cand persoanele conectate la LAN nu depasesc numarul de 100. Clientul se afla pe masina perosnala de rulaj, iar serverul care de cele mai multe ori reprezinta doar baza de date se afla pe o masina server separata.

In acest caz procesarea se realizeaza pe masinile hardware ale clientilor, iar serverul se comporta ca un controler pentru transmisia de date dintre client si baza de date.

Performantele acestui tip de arhitectura sunt slabe, fiind necesara mentinearea unei conexiuni intre server si client pe un timp indelungat, chiar daca nu sunt trimise date. Mentinerea conexiunii ca fiind deschisa este necesara deoarece clientul face mult mai multe cereri catre server pentru achizitia sau modificarea datelor ca acestea sa fie intr-un final afisate utilizatorului

* Three-tier

Aplicatia este separata in trei layere logice, fiecare fiind bine definite si avand un singur scop general.

Primul layer este cel de prezentare care consta in interfata grafica prezentata utilizatorului, prin care acesta poate efectua operatii asupra datelor.

Al doilea layer, denumit “application logic”, se ocupa cu logica propriu zisa a aplicatiei si efectuarea operatiilor de procesare in functie de datele primite de la utilizator prin intermediului layer-ului de prezentare si datele aflate in baza de date.

Al treilea layer, denumit “data layer”, este in principiu baza de date care metine toate datele necesare aplicatiei.

Partea de procesoare fiind executata in cel de al doilea layer reducem semnificativ numarul de cereri din partea Clientului si asta ne aduce o imbunatatire majora a performantelor, implicit a scalabilitati.

De asemenea, layer-ul “application logic” se ocupa de securitatea aplicatiei si de persistenta datelor transmise.

http://norlizakatuk.weebly.com/uploads/2/6/6/0/26606863/saiful\_azmi.pdf